

## REMARKS

Claims 47-86 are pending. Claim 47 and 68 are amended.

The Examiner re-opened prosecution of the present application in the Office Action of June 18, 2004.

The Examiner rejected Claims 47-67 and 68-86 under 35 U.S.C. § 101 as being directed to non-statutory subject matter. With respect to Claims 47-67, the Examiner states, in pertinent part, that "claim 47 is ONLY useful when a computer is not running a claimed method (in claim 47) can not be derived (i.e., merely claiming a floppy disk with claimed instructions) ..." (emphasis in the original). With respect to Claim 68-86, the Examiner states, in pertinent part, that "this claim contains computer-per-se materials, although a processing system is claimed."

As amended, Claim 47 and 68 now recite subject matter that is unambiguously within the technological art of pricing a transaction to allow a customer to be fairly billed for the service or services rendered in the transaction. The Examiner's rejection is believed overcome.

The Examiner repeated his rejection of Claim 47 under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent 5,630,127 ("Moore") in view of U.S. Patent 5,682,482 ("Burt"), stated in the Office Action of Nov. 4, 2003. In pertinent part, the Examiner states:

Moore et al. ('127) disclose that a rule-based application structure could be a relational database where records of a transaction are related/linked to each other (see Moore, the abstract, Figs.3,4). Moore et al. teach that: service instances

linking to transaction instances; creating a billing service instance linked to a service instance with relation instance (see Moore, "FIG. 4 is an object instance table." 6:54-59 "An example of this table is shown in FIG. 3. The names or "objects" are shown in columns "OBJECT" 302, "OBJECT1" 304 and "OBJECT2" 308. These names or "objects" stand for a multitude of particular instances of the data, any of which can be retrieved by specifying the identifiers of the entities listed above which would focus the access on a particular representation value." 10:5-19; 10:45-55 "An additional feature of the GRMS architecture is the placement of GRMS processor on the Business Professional's workstation 118 along with the Object Table 300, and the program defined in the object table 300. Since the object instance table 400 is also present, the Business Professional can change values in the Object instance table (via GRMS screens and functions) and reprocess the report on the workstation. All object accesses will be satisfied by the Object instance table function and therefore, the CMIM database 224 is not needed for this "What if" analysis reporting"; in OOP, "instance" is a variable name e.g., service instance, relation instance etc.).

Although Moore et al. teach about a financial institution, and a single transaction can generate many object instances (see Moore et al., 1:21-30 and Detailed Description Text portion (para. 439) "Unique identifier for a GRMS transaction. A single GRMS transaction can generate many object instances"), Moore et al. do not explicitly disclose that financial transaction functions are connected together.

However, Burt et al. disclose a system with related functions including financial transaction functions connected together (e.g. see Burt et al., Fig. 5, the abstract, 4:25-27, and 25:2-16), comprising:

- creating a transaction instance corresponding to a financial transaction (e.g. see Burt et al., Fig. 5, the abstract, col. 6 lines 1-14, and col. 21 lines 42-59)

The examiner submits that because Moore et al. teach applications using OOP macros wherein "instance" is a variable instance – an instance is a single occurrence of a class -, it would be obvious for the analogous use of macros: "transaction instance", "service instance", and "billing service instance".

Thus, this descriptive material will not distinguish the claimed invention from Moore et al., and Burt et al. In terms of patentability, see *In re Gulack*, 703 F.2d 1381, 1385, 217 USPQ 401, 404 (Fed. Cir. 1983); *In re Lowry*, 32 F.3d 1579, 32 USPQ2d 1031 (Fed. Cir. 1994).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to create different instances in the article of manufacture as shown in Moore and Burt et al. because such data does not functionally relate to the substrate of the article of manufacture and merely labeling the data different from that in the prior art would have been obvious. See *Gulack* cited above.

Applicant respectfully disagrees with the Examiner. As explained in Applicant's Supplementary Appeal Brief of March 2, 2004, the portions of Moore relied upon by the Examiner for his rejection, i.e., Figs. 3 and 4, the abstract and cols. 4-10, merely discloses a rule-based system for currency trading using a relational database, as clearly shown in Figs. 3 and 4. Thus, the following limitations of Claim 47 are neither disclosed nor suggested by Moore:

creating a first production service instance representing an action performed to process said transaction, said first production service instance being linked to said transaction instance by a first relation instance; and

creating a billing service instance representing a billing service related to a pricing of said first production service, said billing service instance being linked to said first production service instance by a second relation instance.

(emphasis added)

Burt also neither discloses nor suggests the above-quoted limitations of Claim 47. In the portions of Burt that the Examiner relied on for his rejection (i.e., Fig. 5, the abstract, cols. 4, 6, 21 and 24-25), Burt merely discloses "a network 10 is provided that includes a number of support system 14." (col. 4, lines 42-43). Figure 5 "summarizes certain that is required by

the agents of the four network layers, as applied to billing functions.” (col 24, lines 56-60). The functions are managed by management, fulfillment, charging and booking “agents.” (col. 21, lines 42-59; col. 24, line 60 to col. 25, line 16). Thus, the combined teachings of Moore and Burt do not disclose or suggest Applicant’s Claim 47.

The Examiner’s assertion that Moore’s teaches “applications using OOP macros wherein “instance” is a variable instance – an instance is a single occurrence of a class -, it would be obvious for the analogous use of macros: “translation instance”, “service instance” and “billing service instance” is irrelevant. Contrary to the Examiner’s assertion, the “instances” recited in Claim 47 are not variable instances of object-oriented program macros. Claim 47 are not limited in any way to the use of objected-oriented programming or any other programming techniques.

Further, the Examiner’s comment regarding the Federal Circuit’s decision in *in re Gulack* is inapposite. As amended and as pointed out above, the differences between Claim 47 and the teachings of Moore and Burt concern the substance of the “first product service instance” and the “billing service instance” and not merely labeling the data differently. *In re Gulack* relates to a “printed matter rejection” that is not relevant to the present claims. Thus, Applicant respectfully submits that Claim 47 is allowable over the teachings of Moore and Burt, whether considered individually or in combination.

The Examiner rejected Claim 68 under 35 U.S.C. § 103(a) as being unpatentable over Moore in view of Burt. To support this rejection, the Examiner cites the same reasons as his rejection of Claim 47. For substantially the same reasons as stated with respect to Claim 47, the combined teachings of Moore and Burt also neither disclose nor suggest Claim 68. Thus,

Claim 68 is allowable over the teachings of Moore and Burt, whether considered individually or in combination.

In paragraphs 9-18, the Examiner rejected Claims 48-54, 56-66, 69-75 and 77-85 under 35 U.S.C. § 103(a) as being unpatentable over Moore in view of Burt. In each of these rejections, the Examiner's rationale for his rejection is not clear. It appears that the Examiner believes without support that the combined teachings of Moore and Burt render obvious any and all financial transactions implemented in a relational database simply because object-oriented programming (OOP) and connected "instances" are taught by Moore and Burt.

Applicant respectfully traverses the Examiner's rejections of Claims 48-54, 56-66, 69-75 and 77-85. The Examiner's reasons are inadequate. In addition to the limitations of their parent Claim 47, which is allowable over Moore and Burt, as explained above, Claims 48-54 and 56-66 recite "second production service instance", "second relation instance", "third relation instance", "second billing service instance", "account instance", "fourth relation instance", "entity instance table", "relation instance table", "settlement service instance", "price table instance", "cost table instance", "fee table instance", and "mandatory relation instance", none of which are disclosed or suggested by Moore or Bert. As explained in Applicant's Specification, at page 25, lines 3-12, for example, using the recited structures allows pricing of complex financial transactions. Neither these structures nor their attendant benefits are disclosed nor suggested by Moore or Bert. Accordingly, Applicant respectfully submits that Claims 48-54 and 56-66 are each allowable over the teachings of Moore and Bert, whether considered individually, or in any combination. For substantially the same

reasons, Claims 69-75 and 77-85 are allowable over Moore and Burt, whether considered individually and in combination.

The Examiner also repeated his rejection of Claim 55 under 35 U.S.C. § 103(a) as being unpatentable over Moore, Burt and U.S. Patent 5,636,117 ("Rothstein"), stated in the Office Action of November 4, 2003. The Examiner refers to his rejection of Claim 54 and further states:

Rothstein further teaches that a market segment instance could be an entity instance (see 2:8-10; 2:54-47; 3:9-12) (e.g., mortgage entities are linked to business models by indices in a program).

Therefore it would have been obvious to one of ordinary skill in the art at the time of the invention to combine specific applications of Moore et al., and Burt et al. in OOP financial transaction with Rothstein because they all suggest a systematic method that use "instance" in OOP to track components of costs and fees each time a financial transaction is processed. Artisan would recognize that an instant in OOP would be a variable to measure the impact of any changes from financial transactions by tracking those instance variables.

Applicant respectfully disagrees with the Examiner. Claim 55 depends from Claims 47, 52 and 54 and thus is allowable for at least the reasons set forth above with respect to Claim 47. In addition, as recited in Claim 55, the market segment instance is linked to the transaction instance by a fourth relation instance, which is neither disclosed nor suggested by any of Moore, Burt and Rothstein. Thus, Claim 55 is allowable over Moore, Burt and Rothstein, individually and in combination. Applicant respectfully requests the Board to reverse the Examiner's rejection of Claim 55 under 35 U.S.C. § 103(a) over Moore, Burt and Rothstein.

The Examiner rejected Claims 64, 66 and 85 under 35 U.S.C. § 103(a) as being unpatentable over Moore, Burt and U.S. Patent 5,559,313 ("Claus"), the Examiner refers to his rejection of Claim 84 and further states:

Claus et al. further express analogous instances in a database, the examiner submits that since they are considered as variable instances in OOPs (see Figs 6, 9-11, 13, 15) for analogous examples that were claimed about:

- an entity instance could be an account instance;
- an entity instance could be a client instance;
- an entity instance could be a market segment instance.

Therefore it would have been obvious to one of ordinary skill in the art at the time of invention to combine certain applications to combine Moore et al., Burt et al., and Claus et al. in financial transaction with OO programming (or different applications using relational database) because they all suggest a systematic method that use "instance" in a structural database to track all the components of costs and fees each time a financial transaction is processed. It has been recognized that a financial system would be able to measure profitability in a flexible manner and to measure the impact of any changes from banking clients by tracking those variables,

Applicant respectfully disagrees with the Examiner. Claim 64 depends from Claims 47 and 63, Claim 66 depends from 47 and 65, and Claim 85 depends from 68 and 84, and thus each of Claims 64, 66 and 85 are allowable for at least the reasons set forth above with respect to Claim 47. In addition, as recited in Claims 64, 66 and 85, the account instance, the client instance, and the market segment instance, respectively, are each linked to the transaction instance by an entity instance, which is neither disclosed nor suggested by any of Moore, Burt and Claus. Thus, Claims 64, 66 and 85 are each allowable over Moore, Burt and Claus, individually and in combination.

In paragraphs 21-32, the Examiner commented on Claims 56-63, 65-67 and 86. The Examiner's comments are not clear as to whether they are supplementary to comments already made in other paragraphs or they are independent additional grounds of rejection. In any event, the Examiner's comments do not address the lack of teachings by Moore, Burt, Rothstein or Claus regarding the limitations in Claims 56-63, 65-67 and 86. Thus, Claims 56-63, 65-67 and 86 are each allowable over the teachings of Moore, Burt, Rothstein or Claus, whether considered individually or in combination.

In his conclusion, the Examiner states:

The examiner submits that the reasons for rejection are obvious vs. cited prior arts. Applicant is suggested to indicate **in the claims how** the claims distinguish from the combining of cited prior arts. An instance, as defined, is a familiar component in object-oriented programming in relation to the class to which it belongs; a definition for "instance variable": a variable associated with an instance of a class (an object); if a class defines a certain variable, each instance in the class has its own copy of that variable. Hence, there is nothing inventive in defining/creating different instances that linking together in a data structure (the definition is already established for an obvious use of "instance" in cited prior art, i.e., an instance is a single example of occurrence of a class).

**Moore et al., Burt et al., and Rothstein et al.** teach applications using object-oriented program (OOP) wherein "instance" are variable instances of a program (please note that "instance" is a macro instruction (an OO programming language, it is a "reserved term" that defines a set of instructions that are substituted for the macro name wherever the name appears in a program. Macros are similar to functions in that they can take argument – an instance is a single example or occurrence of a class, e.g., programming instruction to show **Class vs. Instance...**

**Burt et al.** teach a method with related functions including financial transaction functions utilizing instance, although not expressly disclosed claimed phrase of: "creating



related records to allow pricing transaction” ...; therefore, **Burt** et al. teach analogous functions as claimed (i.e., defining names for a specific application which does not change a required function in programs). **Moore** et al. further clarify such application with rule-based application structure could be a relational database where records of transaction are related to each other ... The examiner submits that “production service instance” and “billing service instance” in claims merely are examples of financial services OOP software using instance as a macro ... these are financial services that have “linking” with each other by the use of a relational database management system disclosed by **Moore** et al.

... The examiner submits that **Burt** et al. teach a relational model as claimed (see **Burt** et al., Fig. 5); “relational model is a data model in which the data is organized in relations (tables)...

... The examiner also submits that suggestions for “Categorization of purchased items for each transaction by a smart card” has been discussed by **Claus** et al. in their patent... and a relational database of items/instances for different transactions were done in a financial software program.

... The act of linking data/items from different parts in a database is in cited references of **Moore** et al., **Burt** et al., **Rothstein** and **Claus** et al., and they are a fundamental knowledge in database structure of OOPs; from that available computer programming knowledge the applicant uses it to apply for a specific use (i.e., for pricing transactions). Therefore, the invention does not teach any new inventive concept according to the cited references.

... The examiner submits that cited prior art’s limitations are not necessary spelled-out exactly claimed languages; analogous interpretations based on definition for function of those terms show that such claimed language would be obvious for meaningful modifications in OOP using in cited arts situations.

All claimed limitation have been known since events for pricing transactions always “link” to related objects in a relational database. As the examiner presents that the claimed subject matter is obvious with one of skills in the art, different “instances” in above claims may be defined according to the use of a particular “instance” in an object oriented program, in

relation to the “class” to which it belongs; in other words, instance variable is just a variable associated with an event/action/instance of a class in OOPs...

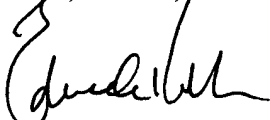
From the Examiner’s lengthy discussion above, it is clear that the Examiner recognizes that the various “production service instance”, “billing service instance” “relation instance”, “entity instance” and other specific instances recited in Applicant’s claims are neither disclosed nor suggested by any of the cited references. The Examiner is well aware that Moore, Burt, Rothstein and Claus at best suggest general notions of financial transaction, and not at all the specific data structures recited in Applicant’s claims. Nevertheless, the Examiner is fixated on the concepts of macros, classes, instances in object-oriented programming language used in some relational database systems. Notwithstanding that such programming concepts are neither recited nor inherent in Applicant’s claims, the Examiner persists in alleging, but fails to demonstrate how, general notions of financial transactions and these concepts of object-oriented programming language render obvious Applicant’s claims, which recite specific instances created in a database specific to solving the problem of pricing transactions involving complex components. The Examiner’s allegations are misplaced. None of Applicant’s claims is limited by any structure of any programming language, object-oriented or otherwise. The combination of general notions of financial transactions and specific programming techniques simply does not disclose or suggest specific data structures necessary for pricing complex transactions. To show that Applicant’s claims are obvious in light of the cited prior art references, the Examiner must show, from the teachings of the prior art itself, how the general notions of financial transactions of Moore, Burt, Rothstein and Claus and the programming language structures may be modified by one of ordinary skill in the art to obtain the specific data structures recited in Applicant’s claims and how these

specific claims can be utilized to price complex financial transactions. The Examiner simply fails to provide such a showing.

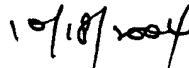
Applicant has explained numerous times since February 2003, including in both an Appeal Brief and a Supplementary Appeal Brief, how the Examiner's rejections are misplaced and erroneous. Nevertheless, the Examiner persists in his same rejections, even re-opening prosecution twice from an appeal before the Board of Appeals and Interferences. The Examiner's actions have caused a significant delay in the prosecution of these claims. The Examiner is respectfully requested to either allow the pending claims or re-instate Applicant's appeal, so that the Board of Appeals and Interferences may rule without further delay on the patentability of Applicant's claims and whether the Examiner's rejections are reasonable.

Applicant respectfully requests allowance of Claims 47-86. If the Examiner has any questions regarding the above, the Examiner is requested to telephone the undersigned Attorney for Applicant at 408-392-9350.

I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope addressed to: Mail Stop Amendment, Commissioner for Patents, P.O. Box 1480, Alexandria, VA 22313-1450 on October 18, 2004.



Attorney for Applicant



Date of Signature

Respectfully submitted,



Edward C. Kwok  
Attorney for Applicant  
Reg. No. 33,938

LAW OFFICES OF  
ACPERSON KWOK CHEN &  
HEID LLP

1762 Technology Drive  
Suite 226  
SAN JOSE, CA 95110  
TEL (408) 392-9250  
FAX (408) 392-9262